

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings of claims in the application:

**Listing of Claims:**

1                   1.       (Currently amended) A system including one or more computer systems  
2       executing one or more computer programs for object model design and validation, the system  
3       comprising:  
4                   a client interface module communicatively coupled to a client device configured  
5       to receive user input and provide a user interface to a user;  
6                   a database configured to store:  
7                   objects corresponding to an object model, and  
8                   metadata objects describing aspects of the object model during design of  
9       the object model, the metadata objects including information used to represent a collection of  
10      objects corresponding to the object model representing model classes, an object used to represent  
11      a single attribute of an object corresponding to the object model representing a model class, an  
12      object used to represent an association between two objects corresponding to the object model  
13      representing model classes, or an object used to represent one end of an association between two  
14      objects corresponding to the object model representing model classes;  
15                  a configuration management module configured to create a deployable collection  
16      of metadata objects from the metadata objects stored in the database, wherein the deployable  
17      collection of metadata objects represents a tree of metadata objects starting at a root metadata  
18      object; and  
19                  a validation engine for validating the metadata objects stored in the database by  
20      confirming the metadata objects comply with one or more validation rules, wherein said  
21      validation engine is configured to:  
22                  perform completeness validation on the deployable collection in response  
23      to a user entered command to perform validation on the deployable collection as a validation

subject to confirm that data associated with the validation subject complies with the validation rules,  
automatically perform correctness validation on the deployable collection when the validation subject is created or updated to confirm that the semantics of the validation subject complies with the validation rules, and  
automatically perform completeness and correctness validation on the deployable collection when requested by the configuration management module.

2-7 (Canceled)

8. (Currently amended) A computer-implemented method for object model design and validation, the method comprising:  
creating, using a processor of a computer system, an instance of a meta metadata object describing aspects of an object model during design of the object model in response to user specified information defining the meta metadata object, the meta metadata object being information used to represent a collection of objects corresponding to the object model representing model classes, an object used to represent a single attribute of an object corresponding to the object model representing a model class, an object used to represent an association between two objects corresponding to the object model representing model classes, or an object used to represent one end of an association between two objects corresponding to the object model representing model classes;  
automatically applying one or more correctness ~~type~~ validation rules using the processor of the computer system to the instance of the meta metadata object upon creation to confirm that the semantics of the instance of the meta metadata object complies with the one or more correctness validation rules;  
[[if]] when a user selects via a user interface validation of the instance of the meta metadata object, applying one or more completeness validation rules using the processor of the computer system to the instance of the meta metadata object to confirm that data associated with the instance of the meta metadata object complies with the one or more completeness validation rules; and

21 automatically applying both the one or more correctness validation rules and the  
22 one or more completeness validation rules using the processor of the computer system to the  
23 instance of the meta metadata object prior to deployment of the instance of the meta metadata  
24 object at runtime.

9. (Canceled)

1 10. (Previously presented) The method of claim 8, wherein the meta metadata  
2 object comprises an object used to represent an association between two objects representing  
3 model classes and wherein applying a validation rule to the instance of the meta metadata object  
4 by the processor includes applying the validation rule to the two objects associated by the  
5 association.

1 11. (Previously presented) The method of claim 8, further including  
2 automatically applying the one or more correctness validation rules using the processor to the  
3 instance of the meta metadata object if the instance of the meta metadata object is automatically  
4 updated or manually updated.

1 12. (Original) The method of claim 11, wherein the meta metadata object is  
2 one of an attribute and an object.

1 13. (Original) The method of claim 8, wherein the meta metadata object is  
2 one of an aggregated collection and a deployable collection.

1 14. (Currently amended) A system including one or more computer systems  
2 executing one or more computer programs for object model design and validation, the system  
3 comprising:

4 a database configured to store:

5 objects corresponding to an object model, and

6 meta metadata objects describing aspects of the object model during

7 design of the object model, the meta metadata objects including information used to represent a

8 collection of objects corresponding to the object model representing model classes, an object  
9 used to represent a single attribute of an object corresponding to the object model representing a  
10 model class, an object used to represent an association between two objects corresponding to the  
11 object model representing model classes, or an object used to represent one end of an association  
12 between two objects corresponding to the object model representing model classes;

13 means for creating an instance of a meta metadata object of the object model in  
14 response to user input; and

15 a validation means for automatically applying one or more correctness ~~type~~  
16 validation rules to the instance of the meta metadata object when the instance of the meta  
17 metadata object is created to confirm that the semantics of the instance of the meta metadata  
18 object complies with the one or more correctness ~~type~~ validation rules, for applying one or more  
19 completeness validation rules to the instance of the meta metadata object if a user manually  
20 selects validation of the instance of the meta metadata object to confirm that data associated with  
21 the instance of the meta metadata object complies with the one or more completeness validation  
22 rules, and for automatically applying both the one or more correctness validation rules and the  
23 one or more completeness validation rules to the instance of the meta metadata object prior to  
24 deployment of the instance of the meta metadata object at runtime.

1 15. (Previously presented) The system of claim 1, further including a  
2 deployment manager to deploy the validated metadata objects during runtime.

1 16. (Previously presented) The method of claim 8, further including:  
2 after applying both the one or more correctness validation rules and the one or  
3 more completeness validation rules, deploying the object instance using the processor during  
4 runtime.

1 17. (New) A computer-readable storage medium storing a computer program  
2 product having instructions executable by a processor of a computer system for implementing  
3 the method of claim 8.